PATENT
Customer No. 49637
Attorney Docket No. PALM-3612

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of:	)	
	)	
Jeffry Harlow LOUCKS	)	Group Art Unit: 2195
	)	
Application No.: 09/965,374	)	Examiner: Jennifer N. To
	)	
Filed: September 26, 2001	)	
	)	Confirmation No. 6414
For: A METHOD OF DYNAMICALLY ENABLING THE EXPANSION OF A COMPUTER OPERATING SYSTEM	)	

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

APPEAL BRIEF UNDER 37 C.F.R. § 1.192

Pursuant to 37 C.F.R. § 1.192, Appellant submits this Appeal Brief to the Board of Patent Appeals and Interferences. In the June 27, 2005 final Office Action, the Examiner finally rejected claims 1-29. A Notice of Appeal having been filed on October 27, 2005, this Appeal Brief is being filed along with a Petition for Extension of Time, a fee payment for a one month extension of time, and the corresponding Appeal Brief fee payment.

Table of Contents

I. Real Party Interest

The real party in interest is PalmSource, Inc., the assignee of the entire right, title, and interest in the application.

II. Related Appeals and Interferences

There are currently no appeals or interferences related to this application that are awaiting decision by the Board of Patent Appeals and Interferences.

III. Status Of Claims

Claims 1-29 are pending. Claims 1-29, as set forth in the Claims Appendix, were rejected in the final Office Action and are involved in this appeal.

IV. Status Of Amendments

An Amendment After Final was filed in the U.S. Patent and Trademark Office on August 24, 2005, subsequent to the final rejection of claims 1-29 in the final Office Action dated June 27, 2005. As indicated in the Advisory Action dated September 2, 2005, the amendments proposed in the Amendment After Final will be entered for purposes of this appeal. Therefore, claims 1-29, as set forth in the Appendix, include the amendments proposed in the Amendment After Final.

V. Summary Of Claimed Subject Matter

The invention set forth in the pending claims relates to methods and systems for dynamically enabling the expansion of a computer operating system. Page 1, lines 6-8. An exemplary computer system 100 includes an address/data bus 110 for communicating information, a central processor 101 coupled with the bus 110 for

processing information and instructions, a volatile memory 102 (e.g., random access memory (RAM), static RAM dynamic RAM, etc.) coupled with the bus 110 for storing information and instructions for the central processor 101 and a non-volatile memory 103 (e.g., read only memory (ROM), programmable ROM, flash memory, EPROM, EEPROM, etc.) coupled to the bus 110 for storing static information and instructions for the processor 101. Page 11, lines 1-8. Exemplary computer system 100 also includes an optional data storage device 104 (e.g., memory card, hard drive, etc.) coupled with the bus 110 for storing information and instructions and an electronic display device 105 coupled to the bus 110 for displaying information to the computer user. Page 11, lines 8-14. Exemplary computer system 100 also includes a signal input/output device 108 which is coupled to bus 110 for providing a communication link between computer system 100 and a network environment. Page 12, lines 8-11.

In one embodiment, a computer operating system 210 is operable on a palmtop computer system (e.g., palmtop computer system 100) and task scheduling mechanism 220 is a kernel, wherein the kernel is responsible for performing essential functions of computer operating system 210. Page 13, lines 4-10. Scheduling mechanism 220 performs the essential function of scheduling processor time for tasks (also known as threads) running on operating system 210. Page 13, lines 12-14. Scheduling mechanism 220 allocates slices of processor time (e.g., processor 101) to a number of tasks, wherein each task (e.g., tasks 230a-230e) is pre-assigned to a slice of time. Page 13, lines 14-16. Scheduling mechanism 220 is not limited to any number of time slices and can schedule its tasks independent of flow control

from any foreground applications or services, including user interface tasks. Page 13, lines 17-21.

Scheduling mechanism 220 gives processing time to each task based on the specific needs of each task. Page 13, lines 23-24. In one embodiment, scheduling mechanism 220 ranks each task based on a variety of indicators, including but not limited to task priority and resource need rating, and based on the ranking, allocates a time slice to each task. Page 13, line 24 to page 14, line 2.

In one embodiment, tasks 230a-230e comprise predetermined tasks (e.g., tasks that are pre-programmed into the operating system) built into the operating system, and generally cannot be dynamically changed by the user. Page 14, lines 4-7. Tasks 230a-230e may comprise any number of tasks, including operating the graphical user interface, accessing a network library, operating an infrared data transfer port, synchronizing a palmtop computer system to a desktop computer system, and operating sound functionality. Page 14, lines 16-18.

In one embodiment of the present invention, the tasks comprise a background thread 250 or service manager, which is allocated a pre-assigned time slice by scheduling mechanism 220. Page 14, lines 13-15. Background thread 250 provides an execution context and data presence for background, system, and interrupt-related activities, and may be scheduled independently of all other foreground tasks. Page 14, lines 16-18.

In one embodiment of the present invention, background thread 250 comprises a service manager for allocating its time slice to a set of dynamically registered services or applications 260 comprised of a dynamic set of applications

that can be altered by third party developers, as opposed to tasks 230a-230e, which cannot generally be altered or changed. Page 14, line 20 to page 15, line 2. As such, third party applications and other applications and services not built into the operating system may receive an execution context from background task 250. Page 15, lines 2-4.

Service manager registered applications 260 generally are background, system, and interrupt services, and may include hardware management such as radios, flash ROM, plug and play, battery chargers and sound, and communication protocols such as IrDA, Bluetooth, TCP/IP listeners and synchronization applications. Page 15, lines 6-10.

A process 300 for scheduling tasks on a computer operating system in accordance with an embodiment of the present invention operates on the computer 15 operating system of a palmtop computer system (e.g., palmtop computer system 100). Page 15, lines 12-16.

The computer operating system cycles through a set of pre-assigned time slices of processor time, with each time slice being associated with a task of the computer operating system, wherein one of the tasks is a background task. Page 15, lines 18-21. The tasks may comprise activities such as operating the graphical user interface, accessing a network library, operating an infrared data transfer port, synchronizing a palmtop computer system to a desktop computer system, and operating sound functionality. Page 15, line 21 to page 16, line 2.

In one embodiment, the computer operating system continues to cycle through the time slices of processing time, with these time slices being allocated by the

scheduling mechanism of the computer operating system. Page 16, lines 4-6. The associated tasks (e.g., tasks 230a through 230e and 250) are all scheduled independently of any foreground task, and the scheduling mechanism can be a kernel for managing essential functions of the computer operating system. Page 16, lines 6-10.

The background thread time slice is allocated its processing time and executes, and the background thread, comprising a service manager, operates as a scheduler for a set of service manager services, thus providing a context for service manager services. Page 16, lines 12-15. The service manager allocates its time slice to a set of registered services, operating similarly to the scheduling mechanism as described above. Page 16, lines 17-19.

In a process 400 for scheduling services on a service manager resident on a background task of a computer operating system, when active and scheduled by the kernel, the service manager searches for code modules identified as a service manager service. Page 16, line 12 to page 17, line 3. In one embodiment, service manager services are code modules comprising both service code and optional extensions libraries, which are dynamically installed, enabled, disabled, and removed. Page 17, lines 3-5.

In one embodiment, the service manager searches the both the volatile memory and non-volatile memory (e.g., RAM 102 and ROM 103) of the computer system for the service manager code modules. Page 17, lines 9-11. Service manager services may include hardware management tools (e.g., radios, flash ROM, battery chargers, and sound) and communications protocols (e.g., infrared data

transfer, Bluetooth, and TCP/IP listeners). Page 17, lines 11-14. Service manager services may be third party applications, and may include interrupt related activities. Page 17, lines 14-15.

The process 400 determines whether the service manager has found a associated service manager service, and if the service manager has not found an associated service, the service manager waits a predetermined period searches again for associated services. Page 17, lines 17-21. If the service manager has found an associated service, the service manager automatically loads (e.g., registers) the service manager service, and by registering the service, the service manager may allocate processing its processing time to the registered service. Page 18, lines 1-4.

In one embodiment of the present invention, the service manager ranks the priority level of the registered service, for example, by considering the specific needs of each task and/or based on a variety of indicators, including but not limited to task priority and resource need rating. Page 18, lines 6-12. When the service manager is allocated its slice of processor time, the service manager allocates an execution presence to a registered service independently of any foreground task that may be operating on the system. Page 18, lines 14-17. In one embodiment, the registered service executed is the least recently used with the highest priority ranking. Page 18, lines 17-19.

In the event that the registered service is a third party application, the service manager operates to allocate an execution presence (e.g., processing time) to the application. Page 19, lines 1-3. Where the registered service is an interrupt service

routine, the service manager further operates to allocate a data presence (e.g., memory) to the interrupt service routine. Page 19, lines 3-5.

VI. Grounds of Rejection to be Reviewed on Appeal

1. **Whether the rejection of claims 1-14 under 35 U.S.C. § 101 should be reversed**

2. **Whether the rejection of claims 1-29 under 35 U.S.C. § 112, second paragraph, should be reversed**

3. **Whether the rejection of claims 1, 3-8, 10, 12-14, 16, 18-22, and 26-29 under 35 U.S.C. § 103(a) based on U.S. Patent No. 6,330,583 to Reiffin ("Reiffin") in view of U.S. Patent No. 6,757,897 to Shi ("Shi") should be reversed**

4. **Whether the rejection of claims 23 and 24 under 35 U.S.C. § 103(a) based on Reiffin in view of U.S. Patent No. 6,098,090 to Burns ("Burns") should be reversed**

5. **Whether the rejection of claim 25 under 35 U.S.C. § 103(a) based on Reiffin in view of Burns and further in view of Shi should be reversed**

6. **Whether the rejection of claims 2, 9, 11, 15, and 17 under 35 U.S.C. § 103(a) based on Reiffin in view of Shi and further in view of Burns should be reversed**

VII. Arguments

A. Summary of the Arguments

With regard to the claim rejection under 35 U.S.C. § 101, the rejection should be reversed because claims 1 and 10 have been amended to recite statutory subject matter, for example, a computer-implemented method.

With regard to the claim rejections under 35 U.S.C. § 112, second paragraph, the rejections should be reversed because the appropriate claims have been amended to resolve the antecedent basis issues and allegedly unclear language identified by the Examiner.

With regard to the claim rejections under 35 U.S.C. § 103(a) based on Reiffin in view of Shi and based on Reiffin in view of Shi and further in view of Burns, the

rejections should be reversed because, as acknowledged by the Examiner, Reiffin does not disclose, teach, or suggest each and every feature recited in the claims and because the Examiner has not demonstrated the requisite motivation for combining the alleged teachings of Shi with Reiffin. Instead, the problems addressed and resolved by Reiffin and Shi are opposite to one another and thus teach away from any such combination.

With regard to the claim rejection under 35 U.S.C. § 103(a) based on Reiffin in view of Burns, the rejection should be reversed because Reiffin fails to disclose or suggest dynamically updating a set of registered services, as recited in independent claim 23. Burns also fails to disclose or suggest such a feature. Instead, Burns discloses a background processor that manages a background task between and pending task structure and an active task structure. This disclosure differs from the recitation in claim 23 of cycling through a set of pre-assigned time slices to schedule tasks that include a foreground task and a background task, wherein the scheduling of the background task is independent from the scheduling of the foreground task. Consequently, Burns fails to overcome the previously-stated shortcomings of Reiffin

With regard to the claim rejection under 35 U.S.C. § 103(a) based on Reiffin in view of Burns and further in view of Shi, the rejection should be reversed because Reiffin fails to disclose or suggest dynamically updating a set of registered services, as recited in independent claim 23, and Burns fails to overcome this shortcoming of Reiffin, as discussed in the preceding paragraph. In addition, the Examiner has not demonstrated the requisite motivation to combine the teachings of Shi with Reiffin, as mentioned above.

B. Detailed Arguments

1. The rejection of claims 1-14 under 35 U.S.C. § 101 should be reversed because the claims recite statutory subject matter

Independent claims 1 and 10 have been amended in the Amendment After Final filed on August 24, 2005 to recite a computer-implemented method in their preamble. Therefore, Appellant submits that claims 1 - 14 conform to §101 and recite statutory subject matter. Accordingly, the rejection under § 101 should be reversed.

2. The rejection of claims 1-29 under 35 U.S.C. § 112, second paragraph should be reversed because the claims are clear and definite

The issues of antecedent basis in claims 1, 10, and 16; 3 and 12; and 5 - 7, 11, 17 and 20-22 have each been addressed in the proposed amendments contained in the Amendment After Final filed on August 24, 2005. See **Claims Appendix**. Further, claims 23 and 26 have also been amended via the Amendment After Final to clarify the relationship between the computer system and the method for scheduling tasks. See **Claims Appendix**.

Appellant respectfully submits that these claims conform to §112, second paragraph, and that the §112 rejection should be reversed.

3. The rejection of claims 1, 3-8, 10, 12-14, 16, 18-22, and 26-29 under 35 U.S.C. § 103(a) based on Reiffin in view of Shi should be reversed because the Examiner has not demonstrated the requisite motivation to combine the teachings of Shi with Reiffin

a. Reiffin

Reiffin teaches a networked solution for processing large compute-intensive tasks on a distributed parallel basis. The primary idea is to utilize unused workstation

time. Reiffin notes that a distributed system utilizes the otherwise "wasted execution time and computation power of the workstations or personal computers by enabling their CPUs to perform in the background parallelized compute-intensive tasks." Col. 1, lines 61 - 65. The network comprises a plurality of workstations or personal computers each having a pre-emptive multitasking feature where a remote network subtask can be processed on the workstation in the background utilizing the otherwise wasted CPU cycles. The processing is done in the background and does not affect the primary processing because the background processing is done using otherwise "wasted execution time." Reiffin's idea involves breaking a task down into parallel subtasks executed simultaneously on different workstations. A "task" in Reiffin is a compute job such as a science analysis that is too large for a single processor. See Abstract.

b. Shi

Shi is directed to an invention for performing various tasks in a multi-tasking or time sliced environment under "heavy network traffic conditions." Col. 3, lines 25-26, 39-40. The problem identified by Shi is that under such heavy traffic conditions, this "task starvation" occurs which means that a processor is unable to execute lower priority tasks each for a sufficient amount of time to allow each low priority task to effectively perform. Shi is a patent owned by Cisco Technology, Inc., and the computer device on which their invention is applied is typically a "router, switch, hub or the like." Col. 3, line 16. These routers and switches transmit packets along various ports and routes the packets to various destinations. Shi explains that the routers and switches have other tasks they perform besides routing packets of data but that under heavy

traffic conditions, the lower priority tasks (packet routing being a high priority task) sometimes do not get accomplished. Therefore, this "task starvation" condition occurs.

c. The Examiner has not demonstrated the requisite motivation to combine the teachings of Shi with Reiffin

To establish a *prima facie* case of obviousness, the Examiner must meet the three criteria of articulating a motivation or suggestion to combine the references, there must be some reasonable expectation of success and the prior art references must teach or suggest all the claim limitations. Appellant submits that the combination of Reiffin and Shi does not meet the criteria for the reasons set forth below.

Several other principles concerning an obviousness rejection are applicable here. These requirements were mentioned briefly in the previous office action. If the Examiner determines there is factual support for rejecting the claimed invention under 35 U.S.C. 103, the Examiner must then consider any evidence supporting the patentability of the claimed invention, such as any evidence in the specification or any other evidence submitted by the appellant. The ultimate determination of patentability is based on the entire record, by a preponderance of evidence, with due consideration to the persuasiveness of any arguments and any secondary evidence. *In re Oetiker*, 977 F.2d 1443, 24 USPQ2d 1443 (Fed. Cir. 1992). The legal standard of "a preponderance of evidence" requires the evidence to be more convincing than the evidence which is offered in opposition to it. With regard to rejections under 35 U.S.C. 103, the Examiner must provide evidence which as a whole shows that the legal determination sought to be proved (i.e., the reference teachings establish a *prima facie* case of obviousness) is more probable than not. MPEP 2142.

Furthermore, the test for obviousness is what the combined teachings of the references would have suggested to one of ordinary skill in the art, and all teachings in the prior art must be considered to the extent that they are in analogous arts. Where the teachings of two or more prior art references conflict, the Examiner must weigh the power of each reference to suggest solutions to one of ordinary skill in the art, considering the degree to which one reference might accurately discredit another. *In re Young*, 927 F.2d 588, 18 USPQ2d 1089 (Fed. Cir. 1991). MPEP 2143.01. Thus, the MPEP requires that each prior art reference must be considered in its entirety, as a whole, including portions that would lead away from the claimed invention. MPEP 2141.02.

The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990). MPEP 2143.01. Applying these principles, Appellant traverses the conclusion that one of skill in the art would be motivated to combine Reiffin with Shi. The Examiner is correct in concluding that Reiffin fails to teach ranking the registered service and scheduling the registered services in the dedicated pre-assigned time slice. The Examiner however concludes that it would be obvious to combine Reiffin with Shi because Shi's ranking process would improve the efficiency in Reiffin's system by providing a "balanced system that can greatly avoid task starvation…." What we shall see is that these two references are technically different and focus on completely different problems (one deals with wasted CPU cycles and the other deals with heavy CPU usage) and thus mutually exclusive. One of skill in the art would recognize these differences and not find

motivation to combine these patents. In fact, there are substantial reasons supporting a conclusion that these references teach away from one another.

As discussed above, Reiffin discloses a networked solution for processing large compute-intensive tasks on a distributed parallel basis with a primary objective being to utilize unused workstation time. Reiffin notes that a distributed system utilizes the otherwise "wasted execution time and computation power of the workstations or personal computers by enabling their CPUs to perform in the background parallelized compute-intensive tasks." Col. 1, lines 61 - 65. The processing is done in the background and does not affect the primary processing because the background processing is done using otherwise "wasted execution time."

Shi, on the other hand, focuses on an invention for performing various tasks in a multi-tasking or time sliced environment under "heavy network traffic conditions." Col. 3, lines 25-26, 39-40. As discussed above, the problem identified by Shi is that under such heavy traffic conditions, this "task starvation" occurs which means that a processor is unable to execute lower priority tasks each for a sufficient amount of time to allow each low priority task to effectively perform. Shi addresses a problem typically associated with a router, switch, hub or the like because routers and switches have other tasks they perform besides routing packets of data. Shi explains that under heavy traffic conditions, the lower priority tasks (packet routing being a high priority task) sometimes do not get accomplished, thereby leading to this "task starvation" condition. Shi's invention addresses this scenario.

Appellant submits that there are several reasons why there is inadequate (by a preponderance of the evidence) motivation or suggestion to combine Reiffin with Shi.

First, the Examiner states that the motivation is that Shi's (The Examiner stated that it was Waldron in paragraph 7 but Appellant considers this to be an error and assumes that this should be Shi) ranking system would improve the efficiency on Reiffin's system by providing a balanced system that can greatly avoid task starvation. The motivation to address this type of issue found in Shi is not found in Reiffin because, as mentioned above, Reiffin's problem is how to utilize wasted execution time. As set forth by Reiffin, the problem is the exact opposite of a "heavy traffic condition" where tasks cannot be accomplished. Reiffin's invention teaches how to take unused processor time and utilize it efficiently. Therefore, a person addressing the problem of utilizing wasted execution time, as discussed in Reiffin, would not look to Shi for a solution to that problem.

Another reason that urges against the combination of these references is that Reiffin's invention applies to a general purpose personal computer or other standard desktop type computers. Shi, on the other hand, clearly focuses on computing devices that have highly specialized application, such as routers, switches and hubs. Thus, although both patents pertain generally to computing devices, their particular subject matters differ greatly. Accepting the Examiner's position could only lead to the conclusion that any two patents dealing generally with computing devices are combinable to solve any problem, regardless of whether the patents are directed to entirely different subject matter or solve completely different problems. Such a conclusion does not reflect the intent of the patent rules or the state of current case law. Therefore, the difference in computing devices and application between these two

references is another reason that would lead one of skill in the art away from trying to blend their teachings.

In view of the foregoing, Appellant submits that the requisite motivation to combine these references is lacking because one of skill in the art would easily recognize that Reiffin will not approach a "task starvation" problem due to the excess, unused computing time. In a typical computer as taught by Reiffin, primary and secondary priority tasks would simply all be processed because of the excess, unused processor capability. Instead of motivation to combine, Appellant submits that these references teach away from any combination. Certainly Reiffin teaches away from any problem associated with a lack of ability of the computer system to be able to process lower priority tasks or to even approach a condition such as "task starvation".

Therefore, since by a preponderance of the evidence there is insufficient motivation or suggestion to combine Reiffin with Shi et al, Appellant submits that the § 103 rejection of claims 1, 3-8, 10, 12-14, 16, 18-22, and 26-29 based on Reiffin in view of Shi should be reversed.

4. The rejection of claims 23 and 24 under 35 U.S.C. § 103(a) based on Reiffin in view of Burns should be reversed because neither Reiffin nor Burns, alone or in combination, discloses the features of claims 23 and 24

a. Burns

Before discussing the rest of claim 23, we'll cite lines 5 - 19 of column 2 of Burns:

> A first thread having a background task to be executed passes a background task reference to a background processor having a thread which executes background tasks. The background processor includes a pending task structure and an active task structure. The first thread invokes a function, or method, of the background processor to insert the background task reference into the pending task structure. Periodically, the background processor

> examines the pending task structure to determine if background tasks have been registered for execution. Upon determining that one or more references exist in the pending task structure, the background processor moves the references from the pending task structure to the active task structure. The background processor reads the active task structure and executes the background tasks in the same thread via the references. Burns, Col. 2, lines 5 - 19.

Clearly Burns here discusses the pending task structure and the active task structure for management by the background processor.

.

b. Neither Reiffen nor Burns, alone or in combination, discloses or suggest "dynamically updating a set of registered services," as recited in claim 23

Appellant agrees with the Examiner that Reiffin does not teach that the set of registered services may be dynamically updated. However, the Examiner asserts that Burns teaches dynamically updating the set of registered services at col. 2, lines 5 - 19. Appellant respectfully submits that Burns does not teach this feature and therefore claims 23 and 24 are patentable. First, claim 23 recites the step of cycling through a set of pre-assigned time slices to schedule a set of tasks comprising a background task and a foreground task, each of the tasks assigned to one of the time slices wherein scheduling of the background task is independent from the scheduling of the foreground task. These set of tasks comprise the background task and the foreground task that are assigned to time slices. Before discussing the rest of claim 23, we'll cite lines 5 - 19 of column 2 of Burns:

> A first thread having a background task to be executed passes a background task reference to a background processor having a thread which executes background tasks. The background processor includes a pending task structure and an active task

> structure. The first thread invokes a function, or method, of the background processor to insert the background task reference into the pending task structure. Periodically, the background processor examines the pending task structure to determine if background tasks have been registered for execution. Upon determining that one or more references exist in the pending task structure, the background processor moves the references from the pending task structure to the active task structure. The background processor reads the active task structure and executes the background tasks in the same thread via the references. Burns, Col. 2, lines 5 - 19.

Clearly Burns here discusses the pending task structure and the active task structure for management by the background processor. Now, the rest of claim 23 does not discuss tasks but discusses the set of registered services. Appellant respectfully submits that inasmuch as Burns is discussing a first thread and a background processor for processing "background tasks," that this differs from the scheduling of a service manager of a "set of services" recited in claim 23. Notably, claim 23 goes on to claim the step of scheduling execution of a service manager operating on the background thread wherein this step further comprises 1) the service manager scheduling a set of services that are registered therewith for execution within its time slice, wherein the set of registered services may be dynamically updated and 2) the service manager allocating a data presence to each of the set of services registered therewith.

Appellant submits that col. 2, lines 5 - 19 of Burns do not teach the limitation of the scheduling execution of the service manager because the service manager schedules a set of services, the set of registered services being dynamically updated. Appellant submits that the scheduled services are clearly different from background tasks. The background tasks are different features in claim 23 and are not the features

that are "dynamically updated." In Burns, the portion cited by the Examiner relates to the pending and active task structures, which in the context of claim 23 is different from the set of registered services.

Accordingly, Appellant submits that the combination of Reiffin and Burns fails to teach each limitation of claim 23. Therefore, the § 103 rejection of claim 23, and claim 24 depending therefrom, should be reversed.

5. The rejection of claim 25 under 35 U.S.C. § 103(a) based on Reiffin in view of Burns and further in view of Shi should be reversed because the Examiner has not demonstrated the requisite motivation to combine the teachings of Reiffin, Burns and Shi

For the same reasons discussed in section B.4. above, neither Reiffin nor Burns, alone or in combination, discloses or suggests the features of independent claim 23. Moreover, for the same reasons set forth in section B.3. above, there is insufficient suggestion or motivation to combine Reiffin with Shi.

Thus, for at least these reasons, Appellant submits that the § 103(a) rejection of claim 25 based on Reiffin in view of Burns and further in view of Shi should be reversed.

6. The rejections of claims 2, 9, 11, 15, and 17 under 35 U.S.C. § 103(a) based on Reiffin in view of Shi and further in view of Burns should be reversed because the Examiner has not demonstrated the requisite motivation to combine the teachings of Shi with Reiffin

For the same reasons discussed in section B.3. above, there is insufficient suggestion or motivation to combine Reiffin with Shi. Moreover, Burns does not overcome the above-noted deficiencies of Reiffen and is not relied upon by the Examiner for such teachings. Instead, Burns is merely relied upon for its alleged teaching of background task searching for one or more of the at least one registered

services associated therewith. Accordingly, the § 103 rejections of claims 2, 9, 11, 15, and 17 should be reversed.

C. Conclusion

For the reasons given above, the Board of Patent Appeals and Interferences is respectfully requested to reverse the outstanding rejections under 35 U.S.C. §§ 101, 112, second paragraph, and 103(a) so that claims 1-29 may be allowed.

To the extent any extension of time under 37 C.F.R. § 1.136 is required to obtain entry of this Appeal Brief, such extension is hereby respectfully requested. If there are any fees due under 37 C.F.R. §§ 1.16 or 1.17 which are not enclosed herewith, including any fees required for an extension of time under 37 C.F.R. § 1.136, please charge such fees to our Deposit Account No. 50-3102.

Respectfully submitted,

Date: January 27, 2006

By: /Thomas M. Isaacson/

Correspondence Address:
Customer No. 49637

Thomas M. Isaacson
Attorney for Appellants
Reg. No. 44,166
Phone: 410-414-3056
Fax No.: 410-510-1433